

Program studiów**Część A) programu studiów*****Efekty uczenia się**

Wydział prowadzący studia:	Wydział Matematyki i Informatyki
Kierunek na którym są prowadzone studia:	informatyka
Poziom studiów	studia pierwszego stopnia
Poziom Polskiej Ramy Kwalifikacji:	poziom 6
Profil studiów:	ogólnoakademicki
Tytuł zawodowy uzyskiwany przez absolwenta:	licencjat
Przyporządkowanie kierunku do dyscypliny naukowej lub artystycznej (dyscyplin), do których odnoszą się efekty uczenia się:	Dyscyplina: - informatyka (71%) - matematyka (29%) Dyscyplina wiodąca: informatyka
Symbol	Po ukończeniu studiów absolwent osiąga następujące efekty uczenia się:
WIEDZA	
K_W01	ma wiedzę w zakresie matematyki obejmującą podstawy analizy matematycznej, algebry, matematyki dyskretnej, logiki i teorii mnogości oraz metod probabilistycznych i statystyki
K_W02	ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie programowania, algorytmów i złożoności, języków formalnych i automatów, architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych, języków i paradygmatów programowania, grafiki i komunikacji człowiek-komputer, baz danych, inżynierii oprogramowania
K_W03	zna najważniejsze konstrukcje programistyczne oraz pojęcia składni i semantyki języków programowania
K_W04	zna metody i techniki projektowania, analizowania i programowania algorytmów
K_W05	zna najważniejsze struktury danych i wykonywane na nich operacje, zna wybrane algorytmy numeryczne oraz metody obliczeń przybliżonych.
K_W06	ma wiedzę na temat architektury współczesnych systemów operacyjnych i układów cyfrowych (logika układów cyfrowych i reprezentacja danych, architektura procesora, wejście-wyjście, pamięć, architektury wieloprocessorowe)
K_W07	zna niskopoziomowe zasady wykonywania programów
K_W08	zna zasady działania systemów operacyjnych ze szczególnym uwzględnieniem procesów, współbieżności, szeregowania zadań i zarządzania pamięcią
K_W09	ma wiedzę na temat zarządzania informacją, w tym dotyczącą systemów baz danych, modelowania danych, składowania i wyszukiwania informacji
K_W10	ma wiedzę na temat różnych paradygmatów programowania i języków programowania
K_W11	ma wiedzę na temat inżynierii oprogramowania, w tym projektowania (wzorce projektowe, architektura oprogramowania, analiza i projektowanie obiektowe)
K_W12	ma wiedzę na temat technologii sieciowych, w tym podstawowych protokołów komunikacyjnych, bezpieczeństwa i budowy aplikacji sieciowych
K_W13	ma podstawową wiedzę dotyczącą prawnych i społecznych aspektów informatyki, w tym odpowiedzialności
K_W14	zna podstawowe zasady bezpieczeństwa i higieny pracy w zawodzie informatyka

K_W15	zna ogólne zasady tworzenia i rozwoju form indywidualnej przedsiębiorczości, wykorzystującej wiedzę, umiejętności i kompetencje społeczne związane z kierunkiem informatyka
UMIEJĘTNOŚCI	
K_U01	potrafi zastosować wiedzę matematyczną do formułowania, analizowania i rozwiązywania prostych zadań związanych z informatyką
K_U02	potrafi planować swoje uczenie się, potrafi pozyskiwać informacje z literatury, baz wiedzy, Internetu oraz innych wiarygodnych źródeł, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie
K_U03	potrafi pracować indywidualnie i w zespole informatyków, w tym także potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów
K_U04	potrafi porozumiewać się przy użyciu różnych technik w środowisku zawodowym oraz w innych środowiskach, w tym w języku angielskim oraz z wykorzystaniem narzędzi informatycznych
K_U05	potrafi pisać, uruchamiać i testować programy w wybranym środowisku programistycznym
K_U06	umie czytać ze zrozumieniem programy zapisane w języku programowania imperatywnego
K_U07	projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy; wykorzystuje podstawowe techniki algorytmiczne i struktur danych
K_U08	posługuje się przyjętymi formatami reprezentacji różnego rodzaju danych stosownie do sytuacji (liczby, tablice, tekst, obrazy, dźwięk i filmy) pamiętając o ich ograniczeniach, np. związanych z arytmetyką komputera
K_U09	potrafi zainstalować i skonfigurować wybrany system operacyjny oraz nim administrować, w tym instalować potrzebne oprogramowanie
K_U10	opisuje problemy związane z wykonywaniem programów współbieżnych; rozumie mechanizmy synchronizacji procesów
K_U11	potrafi wyjaśnić na czym polega zarządzanie pamięcią w systemach operacyjnych, co to jest hierarchia pamięci, co to jest pamięć wirtualna
K_U12	potrafi skonfigurować prostą sieć (jeden serwer, kilku klientów) i nią administrować z wykorzystaniem stosownych narzędzi
K_U13	potrafi dbać o bezpieczeństwo danych, w tym o ich bezpieczne przesyłanie; posługuje się narzędziami kompresji i szyfrowania danych
K_U14	posiada umiejętność tworzenia prostych, bezpiecznych aplikacji internetowych z wykorzystaniem baz danych
K_U15	potrafi zaprojektować wygodny interfejs użytkownika ze szczególnym uwzględnieniem aplikacji internetowych
K_U16	potrafi stworzyć model obiektowy prostego systemu (np. w języku UML)
K_U17	posiada umiejętność budowy prostych systemów bazodanowych wykorzystujących przynajmniej jeden z najbardziej popularnych systemów zarządzania bazą danych
K_U18	posiada umiejętność wykonania analizy danych liczbowych na poziomie statystyki opisowej z wykorzystaniem jednego ze standardowych pakietów statystycznych
K_U19	potrafi formułować zapytania do bazy danych w wybranym języku zapytań
K_U20	potrafi posługiwać się przynajmniej jednym z naukowych pakietów numerycznych i wykonywać za jego pomocą złożone obliczenia numeryczne
K_U21	ocenia przydatność różnych paradygmatów i związanych z nimi środowisk programistycznych do rozwiązywania różnego typu problemów
K_U22	projektuje oprogramowanie zgodnie z metodyką obiektową
K_U23	potrafi ocenić, na podstawowym poziomie, przydatność rutynowych metod i narzędzi informatycznych oraz wybrać i zastosować właściwą metodę i narzędzia do typowych zadań informatycznych
K_U24	potrafi - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować prosty system informatyczny, używając właściwych metod, technik i narzędzi
K_U25	potrafi wykonać prostą analizę sposobu funkcjonowania systemu informatycznego i ocenić istniejące rozwiązania informatyczne, przynajmniej w odniesieniu do ich cech funkcjonalnych
K_U26	tworzy, ocenia i realizuje plan testowania oprogramowania
K_U27	ma umiejętność posługiwania się przynajmniej jednym z najbardziej popularnych systemów zarządzania wersjami

K_U28	posługuje się wzorcami projektowymi
K_U29	umie posługiwać się co najmniej jednym językiem obcym na poziomie średniozaawansowanym (B2)
KOMPETENCJE SPOŁECZNE	
K_K01	Absolwent jest gotów do: przestrzegania zasad i norm obowiązujących informatyka, w tym norm etycznych, rozumienia znaczenia uczciwości intelektualnej w działaniach własnych i innych osób
K_K02	wypełniania zobowiązań społecznych, służenia swoją wiedzą i umiejętnościami, twórczego myślenia w celu udoskonalania istniejących bądź stworzenia nowych rozwiązań
K_K03	krytycznej oceny swojej wiedzy i dalszego jej doskonalenia z wykorzystaniem różnych źródeł informacji
K_K04	pokonywania trudności stojących na drodze do realizacji założonego celu i systematycznej pracy nad wszelkimi projektami, które mają długofalowy charakter

Część B) programu studiów

Opis procesu prowadzącego do uzyskania efektów uczenia się

Wydział prowadzący studia:	Wydział Matematyki i Informatyki
Kierunek na którym są prowadzone studia:	informatyka
Poziom studiów:	studia pierwszego stopnia
Poziom Polskiej Ramy Kwalifikacji:	poziom 6
Profil studiów:	ogólnoakademicki
Przyporządkowanie kierunku do dyscypliny naukowej lub artystycznej (dyscyplin), do których odnoszą się efekty uczenia się:	Dyscyplina: informatyka (71%), matematyka (29%) Dyscyplina wiodąca: informatyka
Forma studiów:	stacjonarne
Liczba semestrów:	6
Liczba punktów ECTS konieczna do ukończenia studiów na danym poziomie:	180
Łączna liczba godzin zajęć dydaktycznych:	1935
Tytuł zawodowy nadawany absolwentom:	licencjat
Wskazanie związku programu studiów z misją i strategią UMK:	Jednym z trzech aspektów misji Uniwersytetu Mikołaja Kopernika jest nauczanie na poziomie akademickim oraz prowadzenie innych form działalności edukacyjnej i popularyzatorskiej, odpowiadających aktualnym i przyszłym potrzebom i aspiracjom społeczeństwa. Zdobywanie wiedzy łączy z rozwojem kompetencji społecznych. Informatyka jest jedną z ważniejszych dla rozwoju cywilizacyjnego dyscypliną nauki. Program studiów wpisuje się w Strategię Rozwoju Uniwersytetu Mikołaja Kopernika na lata 2021 – 2026, w szczególności w cele operacyjne: II.1.2. Kształtowanie kluczowych kompetencji, w szczególności społecznych i emocjonalnych, a także samoorganizację, twórcze myślenie, przedsiębiorczość oraz kompetencje cyfrowe, II.2.1. Zapewnienie powiązania oferowanych treści kształcenia z działalnością naukową, II.3.2 Zwiększenie praktycznego wymiaru kształcenia w oparciu o zidentyfikowane potrzeby rynku pracy.
Przedmioty/grupy zajęć wraz z zakładanymi efektami uczenia się*	

Grupy przedmiotów	Przedmiot	Zakładane efekty uczenia się	Formy i metody kształcenia zapewniające osiągnięcie efektów uczenia się	Sposoby weryfikacji i oceny efektów uczenia się osiągniętych przez studenta
Grupa I. Matematyczna	Repetitorium z matematyki	<p>Wiedza. Student(ka): posiada podstawowe wiadomości o zbiorze liczb rzeczywistych i jego podzbiorach, w szczególności zna pojęcie pierwiastka, potęgi, logarytmu, wartości bezwzględnej; posiada podstawowe wiadomości o funkcjach jednej zmiennej o wartościach rzeczywistych; zna wykresy i własności funkcji elementarnych: wielomianowych stopnia nie większego niż 2, homograficznych, potęgowych, wykładniczych, logarytmicznych, trygonometrycznych, cyklometrycznych; posiada podstawowe wiadomości o wielomianach zmiennej rzeczywistej.</p> <p>Umiejętności. Student(ka): wykonuje i interpretuje obliczenia arytmetyczne i algebraiczne; prowadzi ściśle rozumowanie matematyczne w kontekście pojęć szkolnej matematyki, w szczególności dowodzi proste własności liczb i funkcji; posługuje się zapisami formalnymi w odniesieniu do własności liczb i funkcji jednej zmiennej; ma pewne doświadczenie w świadomym stosowaniu praw logiki i rachunku kwantyfikatorów w odniesieniu do pojęć matematyki „szkolnej”; potrafi odczytać, zinterpretować i wykorzystać informacje o własnościach funkcji na podstawie jej wykresu; szkicuje wykresy podstawowych funkcji elementarnych oraz ich transformacji,</p>	Ćwiczenia, metody poszukujące	Zaliczenie na ocenę
	Elementy logiki i teorii mnogości	<p>Wiedza. Student(ka): zna podstawową terminologię logiki i teorii zbiorów w zakresie pozwalającym na samodzielną lekturę i analizę prostego tekstu matematycznego,</p> <p>Umiejętności. Student(ka): potrafi stosować prawa rachunku zdań i logiki pierwszego rzędu do budowy opisu (specyfikacji) problemu, jego logicznej analizy, przekształcania i poszukiwania rozwiązania; ma umiejętność poprawnego formułowania różnorodnych problemów w języku matematyki oraz przekazywania matematycznego opisu problemu innym.</p>	Wykład, ćwiczenia, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie ćwiczeń na ocenę.
	Algebra liniowa z geometrią analityczną	<p>Wiedza. Student(ka): 3. zna definicje działań na macierzach, różne metody rozwiązywania układów równań liniowych, podstawowe pojęcia (przestrzeń wektorowa, liniowa niezależność, kombinacje liniowa, baza, wymiar, odwzorowanie liniowe, itp.) i twierdzenia algebry liniowej; rozumie związek algebry liniowej i geometrii przestrzeni euklidesowej,</p> <p>Umiejętności. Student(ka): . wykonuje działania i operacje elementarne na macierzach, stosuje podstawowe algorytmy macierzowe, rozwiązuje układy równań liniowych z wykorzystaniem różnych metod; sprawdza podstawowe własności wektorów w przestrzeniach liniowych; znajduje algorytmicznie bazy specjalnych podprzestrzeni liniowych; rozwiązuje problem wektorów i wartości własnych dla endomorfizmów przestrzeni skończonego wymiarowych, operuje pojęciem przestrzeni niezmienniczych endomorfizmu, rozróżnia podstawowe struktury algebraiczne, ich własności oraz podaje ich przykłady; interpretuje podstawowe pojęcia i</p>	Wykład, ćwiczenia, metody: podające, poszukujące	Egzamin z wykładu po semestrze letnim, zaliczenie ćwiczeń na ocenę w obu semestrach

		problemy geometryczne w języku algebry i stosuje aparat algebry liniowej do ich rozwiązywania.		
	Analiza matematyczna I i II	<p>Wiedza. Student(ka): . zna podstawy rachunku różniczkowego i całkowego funkcji jednej i wielu zmiennych.</p> <p>Umiejętności. Student(ka): . - posługuje się w różnych kontekstach pojęciem zbieżności i granicy; potrafi - na prostym poziomie - obliczać granice ciągów i funkcji, zbadać zbieżność szeregów,</p> <p>- umie wykorzystać twierdzenia i metody rachunku różniczkowego funkcji jednej zmiennej i wielu zmiennych w zagadnieniach związanych z poszukiwaniem ekstremów lokalnych i globalnych oraz badaniem przebiegu funkcji,</p> <p>- umie całkować funkcje jednej zmiennej przez części i podstawienie; umie całkować funkcje wielu zmiennych z wykorzystaniem całek iterowanych i przez zamianę zmiennych; potrafi wyrażać pola powierzchni gładkich i objętości jako odpowiednie całki,</p> <p>- potrafi rozwiązywać podstawowe typy równań różniczkowych zwyczajnych.</p>	Wykład, ćwiczenia, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie ćwiczeń na ocenę.
	Matematyka dyskretna	<p>Wiedza. Student(ka): zna podstawowe twierdzenia teorii liczb: zasadnicze twierdzenie arytmetyki i twierdzenie Eulera; zna podstawowe objekty (permutacje, kombinacje, wariacje) i techniki (metoda bijektywna, wzór włączeń i wyłączeń) kombinatoryczne. Zna podstawowe pojęcia teorii grafów.</p> <p>Umiejętności. Student(ka): - potrafi wykorzystać rozszerzony algorytm Euklidesa do rozwiązywania (układów) kongruencji,</p> <p>- potrafi stosować metodę włączeń i wyłączeń do rozwiązywania zadań o charakterze kombinatorycznym,</p> <p>- potrafi rozwiązywać rekurencje jednorodne o stałych współczynnikach.</p>	Wykład, ćwiczenia, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie ćwiczeń na ocenę.
Grupa II. Metody probabilistyczne i statystyka	Wstęp do statystycznej analizy danych	<p>Wiedza. Student(ka): Zna podstawy rachunku prawdopodobieństwa w stopniu wystarczającym do opisu i analizy eksperymentu losowego i przeprowadzania prostego rozumowania statystycznego. Ma wiedzę na temat podstawowych pojęć oraz etapów badania statystycznego i analizy danych.</p> <p>Umiejętności. Student(ka): Wykorzystuje wiedzę matematyczną do formułowania, analizowania i rozwiązywania prostych zadań probabilistycznych i statystycznych. Potrafi przygotować dane do analizy, wyznaczyć statystyki opisowe i przedstawić dane graficznie oraz przeprowadzić proste rozumowanie statystyczne. Posługuje się w tym zakresie przynajmniej jednym z dostępnych na rynku programów statystycznych.</p>	Wykład, ćwiczenia, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie ćwiczeń na ocenę

		<p>Kompetencje społeczne. Student(ka): Myśli analitycznie, potrafi precyzyjnie określić problem oraz podać metody prowadzące do jego rozwiązania, potrafi przeprowadzić rozumowanie posługując się zasadami logiki. Potrafi pracować ze zbiorami danych, rozumie potrzebę prezentowania danych w sposób zrozumiały dla innych, wykonywania analiz oraz przedstawiania ich wyników w formie przystępnej dla specjalistów z innych dziedzin.</p>		
<p>Grupa III. Metody numeryczne</p>	<p>Wstęp do metod numerycznych</p>	<p>Wiedza. Student(ka): 1. zna i rozumie takie pojęcia arytmetyki komputerowej jak: arytmetyka zmiennopozycyjna, liczby maszynowe, błędy bezwzględne i względne, odejmowanie bliskich wielkości, algorytmy stabilne i niestabilne, uwarunkowania,</p> <p>2. zna podstawowe metody numeryczne algebry i analizy, w tym m.in.: metody znajdowania pierwiastków i rozwiązywania układów równań liniowych oraz metody różniczkowania i całkowania numerycznego.</p> <p>3. rozumie różnicę między obliczeniami symbolicznymi oraz numerycznymi.</p> <p>Umiejętności. Student(ka): 1. umie wykorzystywać programy komputerowe do obliczeń symbolicznych,</p> <p>2. potrafi wykorzystywać programy komputerowe do rozwiązywania równań oraz układów równań,</p> <p>3. umie wykorzystywać programy komputerowe do wykonywania podstawowych działań na macierzach,</p> <p>4. potrafi wykorzystywać programy komputerowe do obliczania pochodnych i całek funkcji jednej lub wielu zmiennych,</p> <p>5. umie zastosować programy komputerowe do analizy przebiegu zmienności funkcji.</p> <p>6. potrafi porównać metody rozwiązywania problemów pod względem dokładności uzyskanych rozwiązań.</p> <p>7. potrafi oszacować błędy uzyskanych rozwiązań.</p>	<p>Wykład, laboratorium, metody: podające, poszukujące</p>	<p>Wykład na zaliczenie na ocenę, zaliczenie laboratorium na ocenę</p>
<p>Grupa IV. Systemy komputerowe</p>	<p>Wprowadzenie do systemów wielozadaniowych</p>	<p>Wiedza. Student(ka): - zna ogólny schemat budowy i organizacji systemu operacyjnego,</p> <p>- rozumie organizację pamięci dyskowych (sieciowy system plików),</p> <p>- zna pojęcie procesu i narzędzia do pracy z procesami,</p> <p>- wie do czego służą zmienne środowiskowe, pliki konfiguracyjne globalne i lokalne,</p> <p>- ma podstawową wiedzę o sieciach komputerowych i protokołach sieciowych (TCP/IP),</p>	<p>Laboratorium, metody: podające, poszukujące</p>	<p>Zaliczenie laboratorium na ocenę</p>

		<ul style="list-style-type: none"> - rozumie mechanizmy szyfrowania danych i zna sposoby bezpiecznej pracy w trybach tekstowym i graficznym na zdalnych hostach, - zna podstawy programowania z użyciem powłok systemu i narzędzi przetwarzania danych (sed, awk, PERL), - zna prawa i obowiązki użytkownika systemu. <p>Umiejętności. Student(ka): - potrafi skonfigurować swoje konto do konkretnych zadań,</p> <ul style="list-style-type: none"> - sprawnie zarządza swoimi zasobami dyskowymi – archiwizacja, szyfrowanie, prawa dostępu, itp. - dba o bezpieczeństwo konta użytkownika i systemu, korzysta z połączeń szyfrowanych (SSH) i tunelowanych (VPN), - pisze i wykorzystuje skrypty powłoki do usprawnienia działań w systemie i przetwarzania plików tekstowych, - potrafi korzystać z podstawowych usług systemu z wykorzystaniem menadżerów okien. 		
	Podstawy systemów operacyjnych	<p>Wiedza. Student(ka): ma wiedzę ogólną z zakresu architektury komputerów i systemów operacyjnych; wie, na czym polega zarządzanie pamięcią w systemach operacyjnych, co to jest hierarchia pamięci, co to jest pamięć wirtualna; rozumie mechanizmy synchronizacji procesów.</p> <p>Umiejętności. Student(ka): - definiuje podstawowe pojęcia związane z systemami operacyjnymi, w tym m.in. architekturę systemu i funkcje jej składowych;</p> <ul style="list-style-type: none"> - klasyfikuje systemy operacyjne (ze względu na ich budowę oraz przeznaczenie); - wylicza algorytmy charakterystyczne dla systemów operacyjnych w tym algorytmy zarządzania procesami, pamięcią i we/wy; - analizuje własności poszczególnych składowych systemu operacyjnego (w tym zarządzanie procesami, pamięcią, we/wy); - wyznacza algorytmy do rozwiązywania klasycznych problemów synchronizacji; - stosuje algorytmy (w tym m. in. szeregowania, wymiany, bankiera) do rozwiązywania konkretnych problemów. 	Wykład, laboratorium, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie laboratorium na ocenę
	Podstawy cyberbezpieczeństwa	<p>Wiedza. Student(ka):</p> <ul style="list-style-type: none"> - definiuje podstawowe pojęcia związane z bezpieczeństwem systemów informatycznych w tym rodzaje zagrożeń oraz ataków, architekturę bezpieczeństwa, kontrolę dostępu; - klasyfikuje zagrożenia i ataki oraz przedstawia ich definicje; - zna podstawowe elementy polityki bezpieczeństwa w organizacji. <p>Umiejętności. Student(ka):</p> <ul style="list-style-type: none"> - wylicza, formułuje oraz ilustruje na przykładach podstawowe zagrożenia i ataki ; analizuje sposoby zabezpieczeń, takie jak IDS/IPS czy zapory sieciowe; 	Wykład z elementami konwersatorium. Interaktywny wykład z pokazem, pobudzający do bieżącej dyskusji zagadnień, zadawania pytań, rozwiązywania konkretnych problemów i	Wykład na zaliczenie na ocenę, zaliczenie laboratorium na ocenę

		<ul style="list-style-type: none"> - rozumie rolę uwierzytelniania i kontroli dostępu w kontekście unikania i likwidowania zagrożeń; - stosuje podstawowe pojęcia związane bezpieczeństwem systemów informatycznych. <p>Kompetencje społeczne. Student(ka):</p> <ul style="list-style-type: none"> - rozumie potrzebę zapewnienia bezpieczeństwa danych w środowisku teleinformatycznym; - potrafi zdobywać i rozszerzać wiedzę związaną z bezpieczeństwem systemów informatycznych; - przekazuje posiadaną wiedzę w zrozumiały sposób. 	<p>wykazywania się aktywnością,</p> <p>Laboratorium</p> <p>metody: podające, poszukujące</p>	
Grupa V. Technologie sieciowe	Programowanie sieciowe	<p>Wiedza. Student(ka): 1. ma wiedzę na temat podstawowych technologii sieciowych,</p> <p>2. zna model OSI oraz podstawowe protokoły komunikacyjne w każdej warstwie tego modelu,</p>	Wykład, laboratorium, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie laboratorium na ocenę
	Administrowanie usługami sieciowymi	<p>3. zna zasady komunikacji między procesami na jednej maszynie i komunikacji między procesami na różnych maszynach,</p> <p>4. zna biblioteki języków programowania używane przy programowaniu aplikacji sieciowych,</p> <p>5. zna podstawowe typy sieci komputerowych oraz urządzeń sieciowych w sieci lokalnej</p> <p>6. zna zaawansowane struktury danych i ich zastosowanie w nowoczesnych implementacjach usług sieciowych,</p> <p>Umiejętności. Student(ka): 1. potrafi zaprojektować aplikację klient – serwer oraz zaprogramować je w stosownym języku programowania,</p> <p>2. potrafi skompilować, skonfigurować oraz nadzorować serwery podstawowych usług sieciowych,</p> <p>3. umie monitorować sieć lokalną oraz kontrolować dostęp do podstawowych serwerów usług sieciowych (serwer www, serwer bazy danych, serwer poczty elektronicznej, serwer zdalnej konfiguracji komputera),</p> <p>4. potrafi zorganizować bezpieczny dostęp do sieci lokalnej oraz ochronę tej sieci przed intruzami,</p> <p>5. potrafi zarządzać komputerem i jego systemem operacyjnym,</p> <p>6. umie zaprojektować i udostępnić przez sieć zasoby multimedialne.</p> <p>7. potrafi zarządzać ruchem w sieci, zarówno statycznie jak i z użyciem dynamicznych protokołów trasowania,</p> <p>8. potrafi zaprojektować i uruchomić Wirtualną Sieć Prywatną z użyciem różnych technologii tunelowania oraz protokołami IPSec,</p>	Laboratorium, konwersatorium, metody: podające, poszukujące	Zaliczenie laboratorium i konwersatorium na ocenę

		<p>9. potrafi stworzyć i administrować zaporami ogniowymi w różnych warstwach modelu OSI,</p> <p>10. potrafi klasyfikować, diagnozować i rozwiązywać problemy związane z bezpieczeństwem sieci,</p> <p>13. potrafi wyszukać w literaturze fachowej konieczne informacje teoretyczne oraz praktyczne konieczne do przygotowania publicznego wystąpienia na temat zagadnień sieciowych.</p> <p>Kompetencje społeczne. Student(ka): 1. jest nastawiony na jak najlepsze wykonanie zadania; dba o szczegóły; jest systematyczny,</p> <p>2. skutecznie przekazuje innym swoje myśli w zrozumiały sposób; właściwie posługuje się terminologią fachową; potrafi nawiązać kontakt w obrębie swojej dziedziny i z osobą reprezentującą inną dziedzinę – potencjalnym użytkownikiem systemu komputerowego lub sieci komputerowej,</p> <p>3. zna i przestrzega zasady i normy obowiązujące informatyków, w tym normy etyczne; rozumie społeczną rolę zawodu informatyka.</p>		
<p>Grupa VI. Algorytmika i programowanie</p>	<p>Podstawy algorytmiki i programowania</p> <p>(możliwość wyboru zaawansowanej grupy laboratoryjnej)</p>	<p>Wiedza. Student(ka):</p> <ul style="list-style-type: none"> - zna podstawowe pojęcia i techniki związane z teorią algorytmów: schemat blokowy, pseudokod, specyfikacja problemu (dane wejściowe i wyjściowe), mechanizmy iteracji i rekurencji, funkcje itp., - zna najważniejsze konstrukcje programistyczne oraz podstawową składnię przynajmniej jednego języka programowania wyższego rzędu (sugerowany strukturalny/proceduralny C++), - zna przykłady algorytmów zaprojektowanych metodami przyrostową, dziel i zwyciężaj, zachłanną, dynamiczną, w tym przykłady algorytmów sortowania i wyszukiwania, - zna podstawowe sposoby reprezentacji liczb w pamięci komputera, - zna i rozumie pojęcia złożoności czasowej i funkcji kosztu algorytmu (programu). <p>Umiejętności. Student(ka):</p> <ul style="list-style-type: none"> - potrafi podać specyfikację problemu oraz odczytywać, analizować i zapisać algorytmy w postaci schematu blokowego i pseudokodu, - implementuje algorytmy, dobiera odpowiednie typy danych (w tym tablice i struktury) oraz stosuje podstawowe metody projektowania algorytmów; implementuje programy korzystające z plików dyskowych, - umie wyznaczyć rząd złożoności prostych algorytmów i zastosować notację asymptotyczną O, Ω, Θ, 	<p>Wykład, laboratorium, metody: podające, poszukujące</p>	<p>Egzamin z wykładu, zaliczenie laboratorium na ocenę</p>

		- umie w podstawowym zakresie stosować wskaźniki w języku programowania, w tym, umie zaimplementować elementarne dynamiczne struktury danych (stos, kolejka, lista).		
Algorytmika i programowanie w języku Python	<p>Wiedza. Student(ka):</p> <ul style="list-style-type: none"> - ma uporządkowaną wiedzę ogólną w zakresie podstawowych algorytmów, ich złożoności i zastosowań; zna podstawowe metody projektowania algorytmów i przykłady algorytmów wykorzystujących te metody, zna podstawowe struktury danych i wykonywane na nich operacje - zna przynajmniej jeden język programowania wyższego rzędu w zakresie podstawowym; zna zasady programowania strukturalnego i proceduralnego; - zna podstawowe zasady programowania obiektowego, zna przynajmniej dwa narzędzia pracy z kodem źródłowym <p>Umiejętności. Student(ka):</p> <ul style="list-style-type: none"> - samodzielnie projektuje/modyfikuje algorytmy w celu rozwiązania problemu; potrafi podać specyfikacje algorytmu i zademonstrować jego działanie; implementuje algorytmy i dobiera odpowiednie struktury danych; analizuje wpływ struktur danych na złożoność programów - potrafi pisać, uruchamiać i testować programy w wybranym środowisku programistycznym; umie czytać ze zrozumieniem programy zapisane w wybranym języku programowania; potrafi pracować z obiektami i projektować programy zorientowane obiektowo; potrafi wykorzystać bibliotekę standardową i złożone struktury danych; potrafi budować hierarchiczną strukturę programu - potrafi zaprojektować algorytm rozwiązujący konkretny problem z danej dziedziny; potrafi pisać kod odporny na błędy - potrafi samodzielnie wyszukiwać informacje w literaturze, także w językach obcych <p>Kompetencje społeczne. Student(ka):</p> <ul style="list-style-type: none"> - zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego kształcenia z wykorzystaniem różnych źródeł informacji, krytycznie ocenia informacje dostępne w Internecie - sumienność i dokładność: jest nastawiony na jak najlepsze wykonanie zadania; dba o szczegóły; jest systematyczny 	Laboratoria są połączeniem metody informacyjno-problemowej z praktyczną realizacją algorytmów w postaci programów w języku Python.	Zaliczenie laboratorium na ocenę	
Algorytmy i struktury danych (możliwość wyboru zaawansowanej grupy laboratoryjnej)	<p>Wiedza. Student(ka): - ma uporządkowaną wiedzę ogólną w zakresie algorytmów, ich złożoności i zastosowań;</p> <ul style="list-style-type: none"> - zna podstawowe metody projektowania algorytmów (rekurencja, metoda dziel i rządź, programowanie z nawrotami, dynamiczne, przyrostowe, algorytmy zachłanne) i przykłady algorytmów wykorzystujących te metody; - zna podstawowe struktury danych i wykonywane na nich operacje (tablice, zbiory, struktury wskaźnikowe, listy, stosy, kolejki, drzewa [w tym kopce, lasy zbiorów rozłącznych i drzewa poszukiwań binarnych] i grafy), 	Wykład, laboratorium, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie laboratorium na ocenę	

		<p>- zna podstawowe pojęcia i techniki związane z teorią algorytmów: specyfikacja problemu (dane wejściowe i wyjściowe), częściowa poprawność programu (niezmiennik pętli), całkowita poprawność programu (zbieżność pętli),</p> <p>- zna najważniejsze konstrukcje programistyczne oraz podstawową składnię przynajmniej jednego języka programowania wyższego rzędu (sugerowany C++ z wykorzystaniem biblioteki standardowej, ale bez implementacji własnych obiektów i szablonów),</p> <p>- zna i rozumie pojęcia złożoności czasowej, funkcji kosztu oraz pełnej funkcji kosztu, złożoności średniej i pesymistycznej oraz kosztu zamortyzowanego</p> <p>Umiejętności. Student(ka): - potrafi podać specyfikacje algorytmów i zapisać algorytmy w postaci pseudokodu oraz schematu blokowego,</p> <p>- umie wyznaczyć rząd złożoności algorytmów;</p> <p>- potrafi uzasadnić poprawność (częściową i całkowitą) prostych programów;</p> <p>- implementuje algorytmy i dobiera odpowiednie struktury danych</p> <p>- analizuje wpływ struktur danych i sposobu implementacji operacji na tych strukturach na złożoność programów;</p>		
	<p>Algorytmy i struktury danych II</p>	<p>Wiedza. Student(ka):</p> <ul style="list-style-type: none"> - zna zaawansowane struktury danych, m.in. dla uporządkowanego multizbioru, drzewa licznikowe, potęgowe (Fenwicka), drzewa przedziałowe, kolejka minimów - zna podstawowe pojęcia teorii sieci oraz podstawowe algorytmy wyznaczające maksymalny przepływ w sieci - zna podstawowe algorytmy geometryczne, m.in. położenie punktu względem odcinka, przynależność punktu do wielokąta, znajdowanie otoczki wypukłej, metoda zmiatania - zna przykłady zastosowań omawianych algorytmów <p>Umiejętności. Student(ka):</p> <ul style="list-style-type: none"> - umie zastosować poznane algorytmy na przykładach - potrafi samodzielnie wyszukać (w internecie lub literaturze) oraz zastosować na odpowiednich przykładach algorytmy rozwiązujące zadane problemy - potrafi zaimplementować poznane algorytmy - potrafi pracować indywidualnie i w zespole informatyków, w tym także potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów <p>Kompetencje społeczne. Student(ka):</p> <ul style="list-style-type: none"> - przekazuje innym swoją wiedzę i przemyślenia w zrozumiałym sposób; właściwie rozumie sformułowania pytań i problemów, poprawnie posługuje się terminologią fachową 	<p>Metody dydaktyczne eksponujące: - pokaz</p> <p>Metody dydaktyczne podające:</p> <ul style="list-style-type: none"> - wykład informacyjny (konwencjonalny) - wykład problemowy <p>Metody dydaktyczne poszukujące:</p> <ul style="list-style-type: none"> - ćwiczeniowa - doświadczeń - klasyczna metoda problemowa - laboratoryjna - projektu - referatu 	<p>Egzamin z wykładu, zaliczenie laboratorium na ocenę</p>

		- rozumie potrzebę ciągłego uczenia się		
	Programowanie I (możliwość wyboru zaawansowanej grupy laboratoryjnej)	<p>Wiedza. Student(ka): zna podstawowe metody projektowania, analizowania i programowania algorytmów.</p> <p>Umiejętności. Student(ka): - potrafi pisać, uruchamiać i testować programy w wybranym środowisku programistycznym, w szczególności w językach C i C++,</p> <p>- umie czytać ze zrozumieniem programy zapisane w języku programowania imperatywnego,</p> <p>- projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy; wykorzystuje podstawowe techniki algorytmiczne i struktur danych.</p>	Laboratorium, metody: podające, poszukujące	Zaliczenie laboratorium na ocenę
	Programowanie II (możliwość wyboru zaawansowanej grupy laboratoryjnej)	<p>Wiedza. Student(ka): - zna zasady programowania obiektowego, pojęcie klasy i obiektu, konstruktory i destruktory, zalety programowania obiektowego i metody oraz przypadki użycia, zasady pracy z obiektami i problemy programistyczne z nimi związane, złożone struktury danych (zbiory, listy, stosy, kolejki, drzewa i grafy), pojęcie wskaźnika, referencji i obiektu; zasady dziedziczenia i hierarchicznej budowy programu, zasady wykorzystania funkcji wirtualnych i zaprzyjaźnionych, zasady wykorzystania szablonów, obsługę i tworzenie wyjątków.</p> <p>Umiejętności. Student(ka): - potrafi pracować z obiektami i projektować programy zorientowane obiektowo,</p> <p>- potrafi wykorzystać bibliotekę standardową i złożone struktury danych,</p> <p>- potrafi budować hierarchiczną strukturę programu,</p> <p>- potrafi tworzyć kod uogólniony z wykorzystaniem szablonów,</p> <p>- potrafi pisać kod odporny na błędy z wykorzystaniem mechanizmu przechwytywania wyjątków.</p> <p>- potrafi stworzyć model obiektowy prostego systemu (np. w języku UML),</p> <p>- ocenia przydatność różnych paradygmatów i związanych z nimi środowisk programistycznych do rozwiązywania różnego typu problemów,</p> <p>- projektuje oprogramowanie zgodnie z metodyką obiektową,</p>	Wykład, laboratorium, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie laboratorium na ocenę
	Programowanie III (Java) lub Programowanie III (C#)	<p>Wiedza. Student(ka): - ma ogólną wiedzę na temat różnych paradygmatów programowania i języków programowania (imperatywny, obiektowy), szczegółowo zna metody projektowania i programowania obiektowego (kapsułkowania i ukrywanie informacji, klasy i podklasy, dziedziczenie, polimorfizm, hierarchie klas).</p>	Laboratorium, metody: podające, poszukujące	Zaliczenie laboratorium na ocenę

	<p>Należy zaliczyć jeden z przedmiotów Programowanie III (Java) lub Programowanie III (C#). Drugi z przedmiotów można wybrać jako przedmiot do wyboru.</p> <p>(grupy laboratoryjne pracujące w oparciu o różne języki programowania – możliwość wyboru grupy)</p>	<p>Umiejętności. Student(ka): posiada umiejętność korzystania z baz danych i połączeń sieciowych w języku Java bądź C#; korzysta z języków Java lub C# w tworzeniu aplikacji lub apletów.</p> <p>potrafi właściwie utrzymać kod źródłowy pisanych przez siebie programów (przez odpowiedni jego podział, formatowanie, utrzymanie jednolitej i zrozumiałej konwencji nazewnictwa zmiennych, funkcji, struktur, pól i innych składowych programów, zrozumiałych i czytelnych komentarzy, itp.), ponadto posiadał nawyk częstej kompilacji pisanych przez siebie fragmentów kodu, co pozwala mu na wczesne wykrywanie oczywistych błędów w tworzonym oprogramowaniu,</p> <p>potrafi tworzyć czytelne i dobrze udokumentowane programy poprzez właściwy dobór nazewnictwa poszczególnych jednostek programu.</p>		
	<p>Programowanie funkcyjne</p>	<p>Wiedza. Student(ka):</p> <ul style="list-style-type: none"> - definiuje podstawowe paradygmaty programowania i podaje praktyczne przykłady języków, które je realizują; - opisuje główne cechy programowania funkcyjnego, które odróżniają go od innych paradygmatów; <p>Umiejętności. Student(ka):</p> <ul style="list-style-type: none"> - potrafi projektować i implementować algorytmy w języku funkcyjnym; - potrafi programować i analizować rozbudowane projekty w funkcyjnym języku programowania; <p>Kompetencje społeczne. Student(ka):</p> <ul style="list-style-type: none"> - potrafi zdobywać i rozszerzać wiedzę związaną z programowaniem funkcyjnym; 	<p>Laboratorium z elementami konwersatorium. Interaktywne zajęcia, pobudzające do bieżącej dyskusji, zadawania pytań i rozwiązywania konkretnych problemów.</p>	<p>Zaliczenie laboratorium na ocenę</p>

		- przekazuje posiadaną wiedzę w zrozumiały sposób.		
	Inżynieria oprogramowania	<p>Wiedza. Student(ka): - ma wiedzę na temat inżynierii oprogramowania, w tym projektowania (wzorce projektowe, architektura oprogramowania, analiza i projektowanie obiektowe), wykorzystania API, narzędzi i środowisk wytwarzania oprogramowania (narzędzia do analizy wymagań i modelowania, narzędzia do testowania, narzędzia do podglądu kodu, narzędzia do zarządzania konfiguracjami i wersjami oprogramowania), cyklu życia projektu informatycznego, specyfikacji oprogramowania, walidacji i weryfikacji, utrzymywania oprogramowania (refaktoryzacji).</p> <p>Umiejętności. Student(ka): - potrafi ocenić, na podstawowym poziomie, przydatność rutynowych metod i narzędzi informatycznych oraz wybrać i zastosować właściwą metodę i narzędzia do typowych zadań informatycznych,</p> <p>- potrafi wykonać prostą analizę sposobu funkcjonowania systemu informatycznego i ocenić istniejące rozwiązania informatyczne, przynajmniej w odniesieniu do ich cech funkcjonalnych,</p> <p>- tworzy, ocenia i realizuje plan testowania oprogramowania,</p> <p>- ma umiejętność posługiwania się przynajmniej jednym z najbardziej popularnych systemów zarządzania wersjami,</p> <p>- posługuje się wzorcami projektowymi.</p>	Wykład, laboratorium, metody: podające, poszukujące	Wykład zaliczenie na ocenę, laboratorium na zaliczenie
Grupa VII. Matematyczne podstawy informatyki	Teoria języków formalnych	<p>Wiedza. Student(ka): 1. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie programowania, algorytmów i złożoności, języków formalnych i automatów, języków i paradygmatów programowania,</p>	Wykład, ćwiczenia, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie ćwiczeń na ocenę
	Teoria obliczalności	<p>2. zna podstawowe metody projektowania, analizowania i programowania algorytmów (projektowanie strukturalne, rekurencja, złożoność obliczeniowa).</p> <p>Umiejętności. Student(ka): 1. potrafi zastosować wiedzę matematyczną do formułowania, analizowania i rozwiązywania zadań informatycznych,</p> <p>2. potrafi pozyskiwać informacje z literatury, baz wiedzy, Internetu oraz innych wiarygodnych źródeł, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie,</p> <p>3. projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy; wykorzystuje podstawowe techniki algorytmiczne i struktur danych,</p> <p>4. potrafi ocenić, na podstawowym poziomie, przydatność rutynowych metod i narzędzi informatycznych oraz wybrać i zastosować właściwą metodę i narzędzia do konkretnych zadań informatycznych,</p>	Wykład, laboratorium, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie laboratorium na ocenę

<p>Grupa VIII. Programowanie zespołowe</p>	<p>Programowanie zespołowe</p>	<p>Wiedza. Student(ka): 1. zna i potrafi używać zaawansowane struktury danych oraz metody algorytmiczne do rozwiązania problemów informatycznych,</p> <p>2. ma wiedzę o najnowszych rozwiązaniach sprzętowych i programistycznych w zakresie urządzeń sieciowych, urządzeń mobilnych i komputerów.</p> <p>3. ma podstawową wiedzę na temat ryzyka i odpowiedzialności związanej z systemami informatycznymi, zna zasady netykiety, rozumie zagrożenia związane z przestępczością elektroniczną,</p> <p>Umiejętności. Student(ka): 1. potrafi opisywać algorytmy i struktury danych w sposób dostępny dla osoby, która nie jest informatykiem, potrafi zareklamować wynik pracy informatyka,</p> <p>2. potrafi posługiwać się narzędziami wspomagającymi tworzenie i utrzymanie oprogramowania,</p> <p>3. potrafi utworzyć specyfikację projektu informatycznego, po jego realizacji potrafi dokonać jego oceny i zgodności z początkową specyfikacją,</p> <p>Kompetencje społeczne. Student(ka): 1. zna i przestrzega zasad związanych z ochroną własności intelektualnej innych, przestrzega zasad licencjonowania produktów informatycznych,</p> <p>2. nawiązuje i utrzymuje długotrwałą i efektywną współpracę z innymi; dąży do realizacji celów zespołu poprzez odpowiednie zaplanowanie i organizację pracy swojej i innych; motywuje współpracowników do zwiększenia wysiłku w celu osiągnięcia założonych celów,</p> <p>3. jest nastawiony na jak najlepsze wykonanie zadania; dba o szczegóły; jest systematyczny.</p> <p>4. w przygotowaniu projektu informatycznego potrafi efektywnie i twórczo współpracować w zespole, w szczególności potrafi brać udział w zaplanowaniu i podziale zadań w zespole oraz potrafi właściwie ocenić pracę swoją i innych członków zespołu,</p> <p>5. potrafi aktywnie brać udział w dyskusjach nad projektem, używa fachowej terminologii, potrafi porozumieć się z fachowcem z innej dziedziny wiedzy czy gospodarki,</p> <p>6. potrafi terminowo wywiązywać się z nałożonych na niego zadań, rozumie i przestrzega zasad pracy w grupie.</p>	<p>Laboratorium, metody: podające, poszukujące</p>	<p>Zaliczenie laboratorium na ocenę</p>
<p>Grupa IX. Grafika komputerowa</p>	<p>Podstawy grafiki komputerowej</p>	<p>Wiedza. Student(ka): 1. ma podstawową wiedzę dotyczącą programowania algorytmów grafiki komputerowej,</p> <p>2. rozpoznaje i rozróżnia najważniejsze formaty graficzne i ich reprezentację w pamięci komputera.</p> <p>Umiejętności. Student(ka): 1. implementuje algorytmy rasteryzacji prymitywów geometrycznych,</p>	<p>Laboratorium, metody: podające, poszukujące</p>	<p>Zaliczenie laboratorium na ocenę</p>

		<p>2. rozróżnia i implementuje modele przestrzeni barwnych,</p> <p>3. opisuje w języku macierzy afiniczne przekształcenia geometryczne,</p> <p>4. identyfikuje i tłumaczy podstawowe algorytmy tekstuowania,</p> <p>5. objaśnia sposób rzutowania przestrzennego na płaszczyznę ekranu,</p> <p>6. charakteryzuje i implementuje modele oświetlenia sceny trójwymiarowej.</p>		
Grupa X. Bazy danych	Bazy danych	<p>Wiedza. Student(ka): 1. ma uporządkowaną wiedzę ogólną z relacyjnych baz danych,</p> <p>2. zna sposoby zapobiegania anomalii przy współbieżnym wykonywaniu transakcji,</p> <p>Umiejętności. Student(ka): 1. umie stworzyć w języku encji i związków projekt konceptualnej bazy danych i przekształcić go w model relacyjny,</p>	Wykład, laboratorium, metody: podające, poszukujące	Egzamin z wykładu, zaliczenie laboratorium na ocenę
	Projekt bazodanowy	<p>2. potrafi formułować zapytania do bazy danych w języku SQL,</p> <p>3. potrafi dbać o bezpieczeństwo danych, w tym o ich bezpieczne przesyłanie,</p>	Laboratorium, metoda poszukująca	Zaliczenie laboratorium na ocenę
Grupa XI. Informatyczne przedmioty do wyboru	informatyczne przedmioty do wyboru (lista ustalana na początku każdego roku akademickiego) – łącznie 18 ECTS	<p>Wiedza. Student(ka): posiada wiedzę z zakresu wybranych nieobowiązkowych przedmiotów informatycznych, wskazaną w sylabusach poszczególnych przedmiotów.</p> <p>Umiejętności. Student(ka): 1. posiada umiejętności wskazane w efektach kształcenia wybranych przedmiotów do wyboru,</p> <p>2. potrafi pozyskiwać informacje z literatury, baz wiedzy, Internetu oraz innych wiarygodnych źródeł,</p> <p>3. potrafi w przystępny sposób przedstawić podstawowe fakty w ramach dziedzin pogłębionych przez wybrane przedmioty do wyboru,</p>	W zależności od wybranych przedmiotów	W zależności od wybranych przedmiotów

<p>Grupa XII. Seminarium dyplomowe i egzamin dyplomowy</p>	<p>Seminarium dyplomowe</p>	<p>Wiedza. Student(ka): ma uporządkowaną wiedzę ogólną w zakresie programowania, algorytmów i złożoności, języków formalnych i automatów, architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych, języków i paradygmatów programowania, grafiki i komunikacji człowiek-komputer, baz danych, inżynierii oprogramowania, zawartą w treściach przedmiotów programu studiów.</p> <p>Umiejętności. Student(ka):</p> <ol style="list-style-type: none"> 1. potrafi zastosować wiedzę matematyczną do formułowania, analizowania i rozwiązywania problemów informatycznych związanych z pracą dyplomową, 2. potrafi pozyskiwać informacje z literatury, baz wiedzy, Internetu oraz innych wiarygodnych źródeł, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie, 3. potrafi pisać, uruchamiać i testować programy potrzebne w pracy dyplomowej w wybranym środowisku programistycznym, 4. potrafi utworzyć opracowanie przedstawiające określony problem z zakresu informatyki i sposoby jego rozwiązania, 5. potrafi w sposób przystępny przedstawić podstawowe fakty teoretyczne związane z programami opisywanymi w pracy dyplomowej. 	<p>Metoda seminaryjna</p>	<p>Zaliczenie seminarium na podstawie referatów, prezentacji i pracy seminaryjnej. Warunkiem uzyskania zaliczenia seminarium jest przygotowanie pracy seminaryjnej pozytywnie ocenionej przez prowadzącego seminarium</p>
---	---------------------------------	---	---------------------------	---

	Przygotowanie do egzaminu dyplomowego i egzamin dyplomowy	<p>Kompetencje społeczne. Student(ka): 1. myśli twórczo w celu udoskonalenia istniejących bądź stworzenia nowych rozwiązań,</p> <p>2. samodzielnie i efektywnie pracuje z dużą ilością danych, dostrzega zależności i poprawnie wyciąga wnioski posługując się zasadami logiki,</p> <p>3. jest nastawiony/a na jak najlepsze wykonanie zadania; dba o szczegóły; jest systematyczny,</p> <p>4. skutecznie przekazuje innym swoje myśli w zrozumiały sposób; właściwie posługuje się terminologią fachową; potrafi nawiązać kontakt w obrębie swojej dziedziny i z osobą reprezentującą inną dziedzinę,</p> <p>5. jest nastawiony/a na nieustanne zdobywanie nowej wiedzy, umiejętności i doświadczeń; rozumie potrzebę ciągłego doskonalenia się i podnoszenia kompetencji zawodowych,</p> <p>6. w pełni samodzielnie realizuje uzgodnione cele, podejmując samodzielne i czasami trudne decyzje; potrafi samodzielnie wyszukiwać informacje w literaturze,</p> <p>7. pracuje systematycznie i posiada umiejętność pozytywnego podejścia do trudności stojących na drodze do realizacji założonego celu; dotrzymuje terminów,</p> <p>8. zna i przestrzega zasady i normy obowiązujące informatyków, w tym normy etyczne związane z ochroną własności intelektualnej i korzystaniem z zasobów internetowych.</p>	Praca własna	Egzamin dyplomowy
Grupa XIII. Zajęcia z dziedziny nauk humanistycznych i dziedziny nauk społecznych	Przedmioty do wyboru, np. z oferty zajęć ogólnouniwersyteckich lub oferowane w ramach innych kierunków studiów	<p>Wiedza. Student(ka): zna zagadnienia objęte wybranym przedmiotem. Rozumie w podstawowym zakresie problematykę i metodykę dyscypliny naukowej, której przedmiot dotyczy.</p> <p>Umiejętności. Student(ka): 1. posługuje się podstawowymi pojęciami dyscypliny naukowej właściwej dla wybranego przedmiotu, 2. dostrzega podobieństwa i różnice między metodami dyscypliny właściwej dla wybranego przedmiotu a metodami informatyki.</p>	W zależności od wybranych przedmiotów	W zależności od wybranych przedmiotów
Grupa XIV. Ochrona własności intelektualnej	Ochrona własności intelektualnej	<p>Wiedza. Student(ka): ma podstawową wiedzę dotyczącą ochrony własności intelektualnej, w tym praw autorskich.</p> <p>Umiejętności. Student(ka): umie posługiwać się informacjami potrzebnymi w działalności naukowej, dydaktycznej lub innej związanej z kierunkiem studiów z zachowaniem praw autorskich i ochroną własności intelektualnej,</p> <p>Kompetencje społeczne. Student(ka): zna i przestrzega zasad i norm obowiązujących informatyka, w tym norm etycznych; rozumie społeczną rolę zawodu informatyka; rozumie i docenia znaczenie uczciwości intelektualnej w działaniach własnych i innych osób.</p>	Wykład, metoda podająca	Zaliczenie

Grupa XV. Zajęcia z wychowania fizycznego	Wychowanie fizyczne	<p>Wiedza. Student(ka): posiada elementarną wiedzę z zakresu kultury fizycznej</p> <p>Umiejętności. Student(ka): posiada umiejętności włączania się w prozdrowotny styl życia i kształtuje postawę sprzyjającą aktywności fizycznej na całe życie</p> <p>Kompetencje społeczne. Student(ka): promuje społeczne i kulturowe znaczenie sportu i aktywności fizycznej oraz pielęgnuje własne upodobania z zakresu kultury fizycznej</p>	Zajęcia praktyczne	Zaliczenie
Grupa XVI. Język angielski	Język angielski	<p>Wiedza. Student(ka): 1. zna odpowiednie struktury gramatyczne i posiada zasób słownictwa języka angielskiego niezbędny do ustnego i pisemnego wypowiedzania się na tematy ogólne oraz związane z kierunkiem studiów.</p> <p>Umiejętności. Student(ka): 1. potrafi przygotować wystąpienia ustne w języku angielskim oraz zaprezentować efektywny komunikat słowny w typowych sytuacjach życia codziennego jak również dotyczących zagadnień związanych z kierunkiem studiów,</p> <p>2. potrafi porozumiewać się przy pomocy różnych kanałów i technik komunikacyjnych na tematy ogólne i związane z kierunkiem studiów,</p> <p>3. rozumie dłuższe wypowiedzi i wykłady na temat związany z kierunkiem studiów oraz większość rozmówców porozumiewających się w języku angielskim podczas krajowych i międzynarodowych spotkań,</p> <p>4. analizuje i interpretuje różnego rodzaju teksty i komunikaty słowne oraz znajduje w nich informacje potrzebne do funkcjonowania w życiu codziennym oraz środowisku akademickim,</p> <p>5. posiada umiejętność przygotowania typowych prac pisemnych dla celów akademickich w zakresie języka ogólnego oraz zagadnień właściwych dla studiowanego kierunku,</p> <p>6. samodzielnie tłumaczy z języka angielskiego na język polski tekst o średnim poziomie trudności związany z kierunkiem studiów.</p> <p>Kompetencje społeczne. Student(ka): 1. stosuje samodzielne strategie uczenia się, kierując się wskazówkami wykładowcy i rozumie potrzebę dalszego rozwijania własnych umiejętności językowych,</p> <p>2. jest przygotowany do funkcjonowania w otoczeniu kulturowo i językowo odmiennym.</p>	Ćwiczenia, metoda poszukująca	Zaliczenie na ocenę; egzamin po ukończeniu kursu
Grupa XVII. Praktyki zawodowe	Praktyka zawodowa	<p>Wiedza. Student(ka): 1. zna ogólne zasady tworzenia i rozwoju form indywidualnej przedsiębiorczości, wykorzystującej wiedzę, umiejętności i kompetencje społeczne związane z kierunkiem informatyka,</p> <p>2. wie, z jakich źródeł zdobyć informacje o ofertach praktyk i wymaganiach pracodawców.</p> <p>Umiejętności. Student(ka): 1. umie wyznaczyć sobie cele zawodowe, określić kwalifikacje zawodowe, które chce nabyć,</p>	Praktyka (4 tygodnie)	Zaliczenie zgodnie z Regulaminem praktyk zawodowych

	<p>2. na podstawie posiadanej wiedzy o rynku pracy umie rozstrzygnąć, w instytucjach jakiej branży powinien uzupełniać wiedzę i doświadczenie zawodowe,</p> <p>3. rozpoznaje problemy, w tym zagadnienia praktyczne, które można rozwiązać w oparciu o wiedzę specjalistyczną zdobytą na uczelni,</p> <p>4. potrafi uczyć się samodzielnie.</p> <p>Kompetencje społeczne. Student(ka): 1. dostrzega potrzebę nieustannego zdobywania nowej wiedzy, umiejętności i doświadczeń; ciągłego doskonalenia się i podnoszenia kompetencji zawodowych,</p> <p>2. zauważa wiele aspektów rzeczywistości, widzi zależności i wyciąga wiarygodne wnioski z posiadanych danych,</p> <p>3. dotrzymuje terminów, konsekwentnie realizuje powierzone mu zadania,</p> <p>4. dba o wysoką jakość efektów pracy; samodzielnie realizuje uzgodnione cele, podejmując czasami trudne decyzje; ulepsza istniejące rozwiązania, proponuje nowe,</p> <p>5. zna i przestrzega zasad i norm etycznych; rozumie i docenia znaczenie uczciwości intelektualnej,</p> <p>6. rozumie wagę umiejętności pracy w zespole oraz czytelnego określenia priorytetów i zadań członków zespołu; dąży do realizacji celów zespołu poprzez: odpowiednie zaplanowanie i organizację pracy swojej i innych, konstruktywne podejście do problemów; rozumie potrzebę koordynacji zadań,</p> <p>7. w zrozumiały sposób wyraża swoje myśli, uważnie słucha tego, co mają do powiedzenia inni.</p>		
	<p>Kompetencje społeczne dla przedmiotów z grup I-V, VII, VIII, X, XI, XIII. Student(ka): zna ograniczenia własnej wiedzy i rozumie potrzebę ciągłego jej uzupełniania i pogłębiania; potrafi myśleć analitycznie; świadomie prowadzi proste rozumowania matematyczne zgodnie z zasadami logiki, dba o szczegóły.</p>		

Szczegółowe wskaźniki punktacji ECTS

Dyscypliny naukowe lub artystyczne, do których odnoszą się efekty uczenia się:

	Dyscyplina naukowa lub artystyczna	Punkty ECTS	
		liczba	%
1.	informatyka	117	71
2.	matematyka	48	29

Grupy przedmiotów zajęć	Przedmiot	Liczba punktów ECTS	Liczba ECTS w dyscyplinie: (wpisać nazwy dyscyplin)*****			Liczba punktów ECTS z zajęć do wyboru	Liczba punktów ECTS, jaką student uzyskuje w ramach zajęć prowadzonych z bezpośrednim udziałem nauczycieli akademickich lub innych osób prowadzących zajęcia	Liczba punktów ECTS, które student uzyskuje realizując zajęcia związane z prowadzoną w uczelni działalnością naukową w dyscyplinie lub dyscyplinach, do których przyporządkowany jest kierunek studiów*****/ zajęcia kształtujące umiejętności praktyczne *****
			informatyka	matematyka	pozostałe			
Grupa I. Matematyczna	Repetitorium z matematyki	4		4			2	
	Elementy logiki i teorii mnogości	6		6			4	6
	Algebra liniowa z geometrią analityczną	11		11			6	11
	Analiza matematyczna I	6		6			4	6
	Matematyka dyskretna	6		6			4	6
	Analiza matematyczna II	6		6			4	6
Grupa II. Metody probabilistyczne i statystyka	Wstęp do statystycznej analizy danych	5		5			4	5
Grupa III. Metody numeryczne	Wstęp do metod numerycznych	4		4			3	4

Grupa IV. Systemy komputerowe	Wprowadzenie do systemów wielozadaniowych	3	3				2	3
	Podstawy systemów operacyjnych	6	6				4	6
	Podstawy cyberbezpieczeństwa	3	3				1	
Grupa V. Technologie sieciowe	Programowanie sieciowe	6	6				4	6
	Administrowanie usługami sieciowymi	4	4				3	4
Grupa VI. Algorytmika i programowanie	Algorytmy i struktury danych	6	6			3	4	6
	Algorytmika i programowanie w języku Python	3	3				2	3
	Podstawy algorytmiki i programowania	6	6			3	4	6
	Algorytmy i struktury danych II	6	6				4	6
	Programowanie I	3	3			3	2	3
	Programowanie II	5	5			4	3	5
	Programowanie III (Java) lub Programowanie III (C#)	4	4			4	2	4
	Programowanie funkcyjne	3	3				2	3
	Inżynieria oprogramowania	3	3				2	
Grupa VII. Matematyczne podstawy informatyki	Teoria języków formalnych	6	6				4	6
	Teoria obliczalności	6	6				4	6
Grupa VIII. Programowanie zespołowe	Programowanie zespołowe	6	6			6	2	6

Grupa IX. Grafika komputerowa	Podstawy grafiki komputerowej	3	3				2	3
Grupa X. Bazy danych	Bazy danych	6	6				4	6
	Projekt bazodanowy	2	2				1	2
Grupa XI. Informatyczne przedmioty do wyboru	informatyczne przedmioty do wyboru (lista ustalana na początku każdego roku akademickiego) – łącznie 18 ECTS	18	18			18	12	18
Grupa XII. Seminarium dyplomowe i egzamin dyplomowy	Seminarium dyplomowe	6	6			6	2	6
	Przygotowanie do egzaminu dyplomowego i egzamin dyplomowy	3	3			3		
Grupa XIII. Zajęcia z dziedziny nauk humanistycznych lub dziedziny nauk społecznych	Przedmioty do wyboru np. z oferty zajęć ogólnouniwersyteckich lub oferowane w ramach innych kierunków studiów	4			4	4	2	
Grupa XIV. Ochrona własności intelektualnej	Ochrona własności intelektualnej	1			1		1	
Grupa XV. Zajęcia z wychowania fizycznego	Wychowanie fizyczne							
Grupa XVI. Język angielski	Język angielski	7			7		3	
Grupa XVII. Praktyki zawodowe	Praktyka zawodowa	3			3	3		
RAZEM:		180	117	48	15	57	107	152
			65%	27%	8%	32%	59%	84%

* załącznikiem do programu studiów jest opis treści programowych dla przedmiotów

Program studiów obowiązuje od semestru zimowego roku akademickiego 2022/2023.